1 Just The Pi-Space Formulas

Quick Reference

In this chapter, I'll show just the Pi-Space Formulas without the derivations. The latest code for these examples can be found at https://sourceforge.net/projects/pispacephysicssoftware/.

1.1 More general form of $E=MC^2$

See Introduction to the Pi-Space Theory for more details on derivation

I'll start with the simplest change which is a modification to the Einstein Mass Energy Equation. We add the constant Pi.

$$E = m\left(\pi c^2\right)$$

This is due to the Square Rule. All we need to do here is add the constant π.

1.2 General Solution (v<<C and v<C) To Kinetic Energy Using Pi-Space

The Pi-Space formula for the general solution to Kinetic Energy in Pi-Space is

$$KEvelocity = m * \left(1 - Cos\left(ArcSin\left(\frac{v}{c}\right)\right)\right)$$

This formula is relativistic and gives the same answers as the Newtonian formula where v<<C and also provides the correct answer where v<C. What's more, you don't need General Relativity.

Example.

Let's convert 20 Miles Per Second into Kinetic Energy and get the Newtonian result and then use the Pi-Space formula. Here I use Mathematica expressions.

Mass = 1,
Newton KE = 0.5*((20)^2) = 200

Miles Per Second 186000
Pi-Space KE = (1 - Cos[ArcSin[20.0/186000.0]])*(186000^2) = 200
Note how we multiply by C^2 to get back to the Newtonian analogue

1.3 General Solution (v<<C and v<C) To Potential Energy Using Pi-Space

The general solution to Potential Energy is to place it equal to the new Kinetic Energy equation.

$$\frac{gh}{c^2} = 1 - Cos\left(ArcSin\left(\frac{v}{c}\right)\right)$$

Therefore we can see that Gravity is non-linear as v<C and linear as v <<C. However, it would be nice to derive a more general acceleration formula based purely on velocity change as Newton did. This is covered as the general solution to acceleration.

1.4 Solving for KE=PE for v/c where v<C and v<<C

$$\frac{v}{c} = Sin\left(ArcCos\left(1 - \frac{gh}{c^2}\right)\right)$$

The values match at the lower relativistic values. Again, we can use Mathematica to plot the values. Note gh/cc must be less <= 1 as it is a relativistic formula.

Table[Sin[ArcCos[1-gh]],{gh,0,1.0, 0.1}]
{0,0.43589,0.6,0.714143,0.8,0.866025,0.916515,0.953939,0.979796,0.994987,1.}

Table[Sqrt[2*gh],{gh,0,1.0,0.1}]
{0,0.447214,0.632456,0.774597,0.894427,1.,1.09545,1.18322,1.26491,1.34164,1.41421}

Placing this is a comparison tables

gh/cc	Newtonian Velocity	**Pi-Space Velocity**
0.0	**0.0**	0.0
0.1	**0.447214**	0.43589
0.2	**0.632456**	0.6
0.3	**0.774597**	0.714143
0.4	**0.894427**	0.8
0.5	**1.0**	0.866025
0.6	**1.09545**	0.916515
0.7	**1.18322**	0.953939
0.8	**1.26491**	0.979796
0.9	**1.34164**	0.994987
1.0	1.41421	1.0

As we can see, the Newtonian Velocity is > C while the Pi-Space solution is 1.0 when PE=1.0C.

Important note:

$$Sin\left(ArcCos\left(1 - \frac{gh}{c^2}\right)\right) = Cos\left(ArcSin\left(1 - \frac{gh}{c^2}\right)\right)$$

So we can represent it this way if we choose. We leave it this way for the example to show how it was derived but it’s possible to use it the other way if preferred.

1.5 Solving for KE=PE Escape Velocity where v<C and v<<C

$$v = Sin\left(ArcCos\left(1 - \frac{\left(\frac{GM}{r}\right)}{c^2}\right)\right) * c$$

Which is the same as

$$v = Cos\left(ArcSin\left(1-\frac{\left(\frac{GM}{r}\right)}{c^2}\right)\right)*c$$

So what we need is the mass of the planet and the radius.

Body	Mass (kg)	Radius (km)
Earth	$5.98 * 10^{24}$	6378
Mercury	$3.30 * 10^{23}$	2439
Venus	$4.87 * 10^{24}$	6051
Mars	$6.42 * 10^{23}$	3393
Jupiter	$1.90 * 10^{27}$	71492
Saturn	$5.69 * 10^{26}$	60268
Uranus	$8.68 * 10^{25}$	25559
Neptune	$1.02 * 10^{26}$	24764
Pluto	$1.29 * 10^{22}$	1150
Moon	$7.35 * 10^{22}$	1738
Ganymede	$1.48 * 10^{23}$	2631
Titan	$1.35 * 10^{23}$	2575
Sun	$1.99 * 10^{30}$	696000

Let's take the example of the Earth using the traditional Newtonian mechanism.

As an example, the mass **M** of the Earth is $5.98 * 10^{24}$ kilograms. The radius **r** of the Earth is 6378 kilometers, which is equal to $6.378 * 10^6$ meters. The escape velocity at the surface of the Earth can therefore be calculated by:

$$\begin{aligned}\mathbf{v_{esc}} &= \mathbf{(2 * G * M / r)^{1/2}}\\ &= \mathbf{(\,2 * (6.67 * 10^{-11}) * (5.98 * 10^{24}) / (6.378 * 10^{6})\,)^{1/2}}\\ &= \mathbf{1.12 * 10^{4}\ meters/second}\\ &= \mathbf{11.2\ kilometers/second\ APPROX}\end{aligned}$$

Mathematica **Sqrt[2*(6.67*10^-11)*(5.98*10^24)/(6.378*10^6)] =** 11183.7

So, let's use the Pi-Space formula.

First point to note is that the Gravitational potential must be expressed in terms of an area change.

So we need to have the speed of light which is 299,792,458 meters per second.

Also, once we have the result, this is an area calculation; we need to convert it back to a velocity so we need to multiply the answer by the speed of light.

This equates to the following Mathematica expression.

Sin[ArcCos[1-((((6.67*10^-11)*(5.98*10^24))/(6.378*10^6))/(299792458^2))]] * 299792458

This produces an answer of 11183.7 meters per second, or 11.1837 kilometers per second.

TODO: Fill out the other planets

Planet	**Mass**	**Radius**	**Newton Escape Velocity**	**Pi-Space E/V**
Earth	$5.98 * 10^{24}$	6378	11183.7	11.1837
Mercury	$3.30 * 10^{23}$	2439		
Venus	$4.87 * 10^{24}$	6051		
Mars	$6.42 * 10^{23}$	3393		
Jupiter	$1.90 * 10^{27}$	71492		
Saturn	$5.69 * 10^{26}$	60268		
Uranus	$8.68 * 10^{25}$	25559		
Neptune	$1.02 * 10^{26}$	24764		
Pluto	$1.29 * 10^{22}$	1150		
Moon	$7.35 * 10^{22}$	1738		
Ganymede	$1.48 * 10^{23}$	2631		
Titan	$1.35 * 10^{23}$	2575		
Sun	$1.99 * 10^{30}$	696000		

1.6 Solving for a Black Hole Event Horizon

$$r = \frac{GM}{c^2}$$

The Newtonian derivation is, where the ‘2’ is due to the averaging of the velocities. Pi-Space does not need to do this as discussed earlier and uses an Integral.

$$r = \frac{2GM}{c^2}$$

Therefore for Earth, the back hole radius is

Newtonian / Schwarzchild derivation
2*(6.67*10^-11)*(5.98*10^24)/(299792458^2) = 0.00887597 = 8.8 mm approx

Versus

Pi-Space derivation
(6.67*10^-11)*(5.98*10^24)/(299792458^2)= 0.00443798 = 4.4 mm approx

1.7 General Solution (v<<C and v<C) To Acceleration

So we need a simple version and a more complex version using Integration for larger velocity ranges.

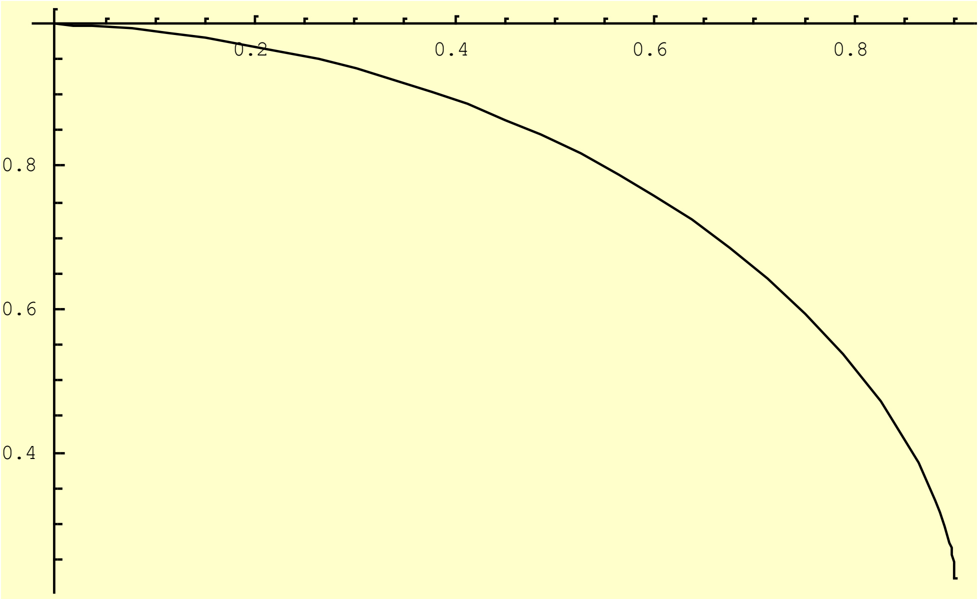
1. For v2/c – v1/c << C use

$$v_2^{'} = ArcSin\left(\frac{v_2}{c}\right)$$
$$v_1^{'} = ArcSin\left(\frac{v_1}{c}\right)$$

$$\alpha = \frac{Sin\left(v_2^{'}\right) - Sin\left(v_1^{'}\right)}{v_2^{'} - v_1^{'}}$$

In Mathmetica, we can plot constant velocity increases of 0.1 to 0.2, 0.2 to 0.3 etc;

Plot[(Sin[ArcSin[x+0.1]]-Sin[ArcSin[x]])/(ArcSin[x+0.1]-ArcSin[x]),{x,0,0.9}];



Table[(Sin[ArcSin[v+0.1]]-Sin[ArcSin[v]])/(ArcSin[v+0.1]-ArcSin[v]),{v,0,0.9,0.1}]

{0.998329,0.988235,0.967729,0.936118,0.892204,0.834012,0.758171,0.658338,0.51955,0.221716}

Velocity 0..V, t=1, constant acc	Newtonian Acc m/s/s	**Pi-Space** α
0.0 to 0.1	**0.5**	0.5 * 0. 998329
0.1 to 0.2	**0.5**	0.5 * 0. 988235
0.2 to 0.3	**0.5**	0.5 * 0. 967729
0.3 to 0.4	**0.5**	0.5 * 0. 936118
0.4 to 0.5	**0.5**	0.5 * 0. 892204
0.5 to 0.6	**0.5**	0.5 * 0. 834012
0.6 to 0.7	**0.5**	0.5 * 0. 758171
0.7 to 0.8	**0.5**	0.5 * 0. 658338
0.8 to 0.9	**0.5**	0.5 * 0. 51955
0.9 to 1.0	0.5	0.5 * 0. 221716

Also for V=C (0.999999C to 1.0)

(Sin[ArcSin[1.0]]-Sin[ArcSin[0.999999]])/(ArcSin[1]-ArcSin[0.999999])= 0.000707107

So there is virtually no acceleration near the speed of light as the Pi-Shell has no remaining area.

2. For v2/c – v1/c < C, you can use an Integral, summing and then averaging all the slopes. You can assume Δx = 0.0001

$$\alpha = \frac{\int_{v_1'}^{v_2'} \frac{Sin(x+\Delta x) - Sin(x)}{\Delta x}}{v_2' - v_1'}$$

e.g. (NIntegrate[((Sin[x]-Sin[x-0.000001])/0.000001),{x,0, 1.5707}])/(1.5707)

Table[((NIntegrate[((Sin[x]-Sin[x-0.000001])/0.000001),{x,i,i+0.1}])/(0.1)),{i,0,1.57,0.1}]

{0.998334,0.988359,0.968509,0.938982,0.900072,0.85217,0.795752,0.731384,0.659709,0.581441,0.497364,0.408318,0.315191,0.218916,0.120453,0.0207867}

1.8 Pi-Space Solution to Einstein's SR Lorenz-Fitzgerald Relativity Formula

$$PiSpaceVelocity = Sin\left(ArcSin\left(\frac{v}{c}\right)\right) = \frac{v}{c}$$

Note: This is the velocity relative to the Newtonian Observer and is equivalent to Newtonian velocity.

$$NewtonianObserver = Hypotenuse = 1$$

Therefore, the relativistic observer is the remaining area of a right-angled triangle. Remember that a right-angled triangle represents Pi-Shell area addition, expressed in terms of the diameter. A Pi-Space rule of thumb is that Cosine represents Pi-Shell compression so we use Cosine for the case where we are losing area.

$$RelativisticObserver = Cos\left(ArcSin\left(\frac{v}{c}\right)\right) = \sqrt{1 - \frac{v^2}{c^2}}$$

This is equivalent to

$$SRLorenzFitzgerald = \sqrt{1 - \frac{v^2}{c^2}}$$

Velocity 0..C	Lorenz-Fitzgerald Sqrt(1-v/c*v/c)	**Pi-Space Cos(ArcSin(v/c))**
0.0	**1.0**	1.0
0.1	**0.994987**	0.994987
0.2	**0.979796**	0.979796
0.3	**0.953939**	0.953939
0.4	**0.916515**	0.916515
0.5	**0.866025**	0.866025
0.6	**0.8**	0.8

0.7	**0.714143**	0.714143
0.8	**0.6**	0.6
0.9	**0.43589**	0.43589
1.0	0.0	0.0

1.9 General Solution to the Average Velocity (v<<C, v<C)

$$\frac{1-Cos\left(ArcSin\left(\frac{v}{c}\right)\right)}{ArcSin\left(\frac{v}{c}\right)}$$

Table[(1-Cos[ArcSin[x]])/(ArcSin[x]), {x,0.1,1, 0.1}]

Velocity 0..C	Newton Average velocity	**Pi-Space Average Velocity**
0.0	**0.0**	0.0
0.1	**0.05**	0.0500418
0.2	**0.1**	0.100339
0.3	**0.15**	0.151171
0.4	**0.2**	0.202871
0.5	**0.25**	0.255873
0.6	**0.3**	0.3108
0.7	**0.35**	0.368659
0.8	**0.4**	0.431362
0.9	**0.45**	0.503773
1.0	0.5	0.63662

1.10 General Solution to Distance an Object Travels as it Accelerates

$$dis\tan ce = v_0 t + \left(\frac{1-Cos\left(ArcSin\left(\frac{at}{c}\alpha\langle v_0, v_0+at\rangle\right)\right)}{ArcSin\left(\frac{at}{c}\alpha\langle v_0, v_0+at\rangle\right)}\right)t$$

Where

$$\frac{at}{c}\alpha\langle v_0, v_0+at\rangle \le 1$$

Note: α is applied to the acceleration range vel *start* to vel *end* e.g. 0.1 to 0.2C

Note: There is no straight-forward way to solve for time t using this approach but it is more accurate while calculating distance.

Table[((1-Cos[ArcSin[0.01*t]])/(ArcSin[0.01*t]))*t,{t,1,10,1}]
{0.00500004,0.0200007,0.0450034,0.0800107,0.125026,0.180054,0.2451,0.320171,0.405274,0.500418}

Versus

Table[(0.5.01*(t*t)), {t,1,10, 1}]*

Time in seconds, acc=0.1m/s/s	Pi-Space Distance	**Newton Distance**
1	**0.00500004**	0.005
2	**.0200007**	0.02
3	**0.0450034**	0.045
4	**0.0800007**	0.08
5	**0.0800107**	0.125
6	**0.180054**	0.18
7	**0.2451**	0.245
8	**0.320171**	0.32
9	**0.405274**	0.405
1.0	0.500418	0.5

1.11 General Solution to the final velocity of a falling Object

$$V_f = Sin\left(ArcCos\left(1 - \frac{ax}{c^2}\alpha\langle v_0, v_0 + ax\rangle - \frac{{v_0}^2}{c^2}\right)\right) * c$$

We can use the Newtonian linear approximation to get the approximate gamma velocity range and the gamma function will make acceleration non-linear. Also we need the Kinetic Energy component of the initial Velocity.

$$V_f = Sin\left(ArcCos\left(1 - \frac{ax}{c^2}\alpha\left\langle v_0, \sqrt{v^2{}_0 + 2ax}\right\rangle - KE\left(\frac{v_0}{c}\right)\right)\right) * c$$

Note: In theory one could apply a shortening to the 'x' value but for this case, we just adjust the acceleration using the gamma mechanism.

Table[(Sqrt[2*x]),{x,0.1,1,0.1}]

{0.447214,0.632456,0.774597,0.894427,1.,1.09545,1.18322,1.26491,1.34164,1.41421}

Table[(Sin[ArcCos[1-x]]),{x,0.1,1,0.1}]

{0.43589,0.6,0.714143,0.8,0.866025,0.916515,0.953939,0.979796,0.994987,1.}

And for smaller values

Table[(Sin[ArcCos[1-x]]),{x,0.000001,.00001,0.000001}]

{0.00141421,0.002,0.00244949,0.00282842,0.00316227,0.0034641,0.00374165,0.00399999,0.00424263,0.00447212}

Table[(Sqrt[2*x]),{x,0.000001,.00001,0.000001}]

{0.00141421,0.002,0.00244949,0.00282843,0.00316228,0.0034641,0.00374166,0.004,0.00424264,0.00447214}

1.12 Distance Travelled at Final Velocity

$$h = \frac{1 - Cos\left(ArcSin\left(V_f\right)\right)}{g}$$

```
Velocity 100.0 MS
Gravity 9.8 MSS

Distance 510 Metres
```

1.13 Solving for time t to travel distance x

Solve[v*t + ((1-Cos[ArcSin[a]])/(ArcSin[a]))*(t*t) == s,t]

$$\left\{\left\{t \to \frac{-v\,\text{ArcSin}[a] - \sqrt{4\,s\,\text{ArcSin}[a] - 4\sqrt{1-a^2}\,s\,\text{ArcSin}[a] + v^2\,\text{ArcSin}[a]^2}}{2\left(1-\sqrt{1-a^2}\right)}\right\}, \left\{t \to \frac{-v\,\text{ArcSin}[a] + \sqrt{4\,s\,\text{ArcSin}[a] - 4\sqrt{1-a^2}\,s\,\text{ArcSin}[a] + v^2\,\text{ArcSin}[a]^2}}{2\left(1-\sqrt{1-a^2}\right)}\right\}\right\}$$

v=0
a=0.01
s=0.405274

Solve[v*t+((1-Cos[ArcSin[(a)]])/(ArcSin[(a)]))*(t*t) == s,t]

{{t→-9.00301},{t→9.00301}} so the solution is 9 seconds

Using traditional Newtonian approach.

Solve[v*t+0.5*a*(t*t) == s,t]

$$\left\{\left\{t \to \frac{1.\left(-1.\,v - 1.41421\sqrt{1.\,a\,s + 0.5\,v^2}\right)}{a}\right\},\ \left\{t \to \frac{1.\left(-1.\,v + 1.41421\sqrt{1.\,a\,s + 0.5\,v^2}\right)}{a}\right\}\right\}$$

```
{{t→-9.00304},{t→9.00304}}
```

1.14 Newton's Gravity Formula

$$Fg = \frac{G^{'}Mm}{\pi . r^{2}}$$

$$G^{'} = G * \pi$$

G' is a modified Universal Gravitational constant. The overall result of the formula is the same but hopefully the reason for it working is more intuitive using this formulation.

This modified value is `2.0963847777404688E-10 and is the Pi-Space Universal Gravitation Constant.`

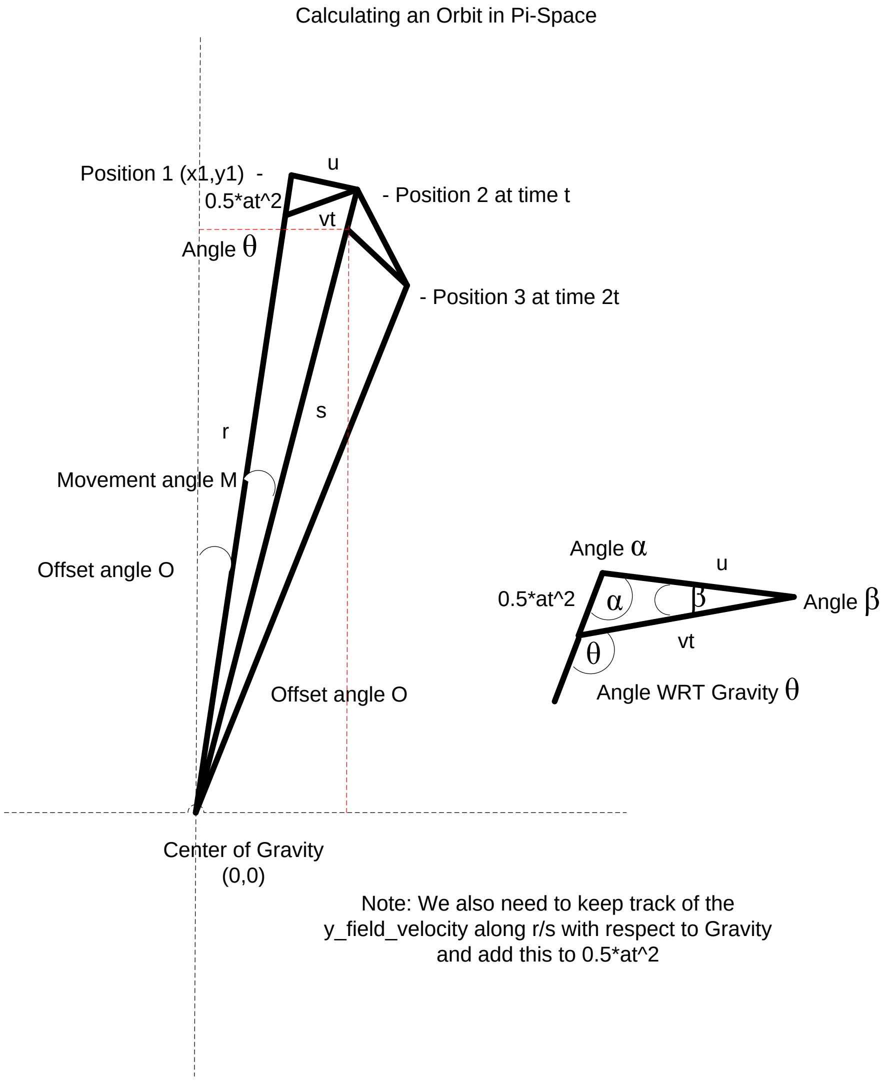
1.15 General Solution to Orbits for Pi Space using Law of the Sines and Law of the Cosines

Typically orbits are covered using Kepler's approach. In Pi-Space, the idea is to come up with a general approach to movement, similar to Newton's Centripetal force idea. In Pi-Space we don't talk about Ellipses or centripetal forces. We talk about adding the force generated by the field with the energy of the moving object. The Law of the Sines and the Law of the Cosines are used to calculate the next position. This is really just a more general form of the Pythagorean Theorem. The idea is that this approach can be used for both trajectories like cannon balls on Earth and the orbits of planets.

We can use the Law of the Cosines and the Law of the Sines to produce an elliptical orbit and the other types of orbits, using these two Laws in Pi-Space. Remember that the position, velocity and time are the Product of Pi-Shell addition. The Law of the Cosines and Law of the Sines are General Pi-Shell addition formulas.

We can use the Law of the Cosines and the Law of the Sines to produce an elliptical orbit and the other types of orbits, using these two Laws in Pi-Space. Remember that the position, velocity and time are the Product of Pi-Shell addition. The Law of the Cosines and Law of the Sines are General Pi-Shell addition formulas.

To calculate the orbit, all one needs to know is the distance from the center of gravity, the velocity of the object and its angle with respect to the center of gravity. This is angle θ.



The high level steps are.

1. Choose x1,y1 moving with velocity v under an acceleration a and angle θ to that Gravity field, center of gravity distance r, offset angle O wrt to axes
2. Calculate the particle velocity x_velocity
3. Calculate the field velocity y_velocity
4. Calculate gamma based on xy_velocity
5. Break out y_velocity into pure y_velocity based on size of gamma
6. Break out particle y_velocity to be added to x_velocity based on (1-gamma)
7. Add particle y_velocity to x_velocity (this will be 0 for small velocities)
8. Calculate a from Newton a = GM/r^2 (M is mass of object), multiply by gamma
9. Calculate the Interior Angle (180 – θ) of orbit triangle
10. From 0.5a*t^2+y_velocity, vt and Interior Angle, calculate u (Law of Cosines)
11. From u, Interior Angle, 0.5a*t^2+y_velocity, calculate β (Law of Sines)
12. Alter the InteriorAngle to handle the Curvature of Space Time
13. OppositeVertex = LawOfCosinesDistance(accelerationDist,paticleDist,innerAngle);
14. Curvature Angle = (1-Gamma)* LawOfSinesAngle(oppositeVertex,InteriorAngle,accelerationDist)
15. InteriorAngle = InteriorAngle - CurvatureAngle

16. Calculate α from 180 – β - InteriorAngle
17. Calculate S from t,u, α (Law of Cosines)
18. Calculate M from s, α,u (Law of Sines)
19. Calculate New Offset Angle = 0 + M
20. Goto step 1, d(new) = s, θ(new) = α, v_x_particle(new) = vt(constant), v_y_field(new)=v_y_current+0.5a*t^2, offset angle O is O+M
21. (new)x1 = s * Cos(90-New Offset Angle), (new)y1 = s * Sin(90-New Offset Angle)

Note: The field and the particle velocity must be adjusted to take into account the relative velocity.

Newtonian acceleration must be multiplied by the gamma function and scaled down to 0 as it approaches speed of light C.

Also, the field velocity tends to 0 at speed of light C and must be added to Particle Velocity.

Note that the Curvature Angle will be typically 0 unless one is travelling at the Speed of Light and Gravity is strong. This is not drawn in the diagram.

Note: See Appendix A for worked Java code implementing this idea which will be a snap-shot.

The latest code can be found on SourceForge https://sourceforge.net/projects/pispacephysicssoftware/

1.16 Bernoulli And Pi-Space

We have Pitot and Venturi Formulas in Pi-Space

Pitot

$$v = Cos\left(ArcSin\left(1 - \frac{\left(\frac{Pt - Ps}{\rho} \right)}{c^2} \right) \right) * c$$

We can calculate the values in the following way

Let’s do a simple calculation to solve for velocity knowing pressure. In Pi-Space, Energy is an area loss of a Pi-Shell. Velocity is a diameter line change.

Pressure is an energy calculation and is therefore an area loss.

We use an imperial system example

Where we have PSI

Let’s take an example where the dynamic pressure is 1.040 lb/ft^2

Also the density of air is 0.002297 slug/ft^3

Using the classic formula, Using Mathematica

Sqrt[2∗(1.04)/(0.002297)] = 30.092 ft/s

Now let’s use the Pi-Space formula

This formula requires that we use the speed of light in feet per second

the speed of light = 983,571,056 foot per second

Sin[ArcCos[1 - (((1.04)/(0.002297))/(983571056^2))]]*983571056 = 29.3127

Now we can see this is not the same as the Classical Result.

The Pi-Space Theory maintains that this is a “more accurate” result than the classical approach.

The Classical Approach is just an approximation.

Let’s make Pi-Space match the Classical approach.

For the speed of light, we set it to 9835710 foot per second (incorrect) instead of 983571056 foot per second

Sin[ArcCos[1 - (((1.04)/(0.002297))/(9835710^2))]]*9835710 = 30.092

Therefore, the more accurate the speed of light calculation, the more accurate the Pitot Velocity result in the Pi-Space Theory.

Note: This would have to be proven/disproven by actual experimentation. I do not have the equipment for this.

Here is a table showing the range of values which are approximate to one another.

```
Table[Sin[
  ArcCos[1 - (((psi)/(0.002297))/(983571056^2))]]*983571056, {psi, 1,
  30, 1}]

{29.3127, 41.4544, 50.7711, 58.6254, 65.5452, 71.8012, 77.5541, \
82.9088, 87.9381, 93.8464, 98.3178, 102.594, 106.7, 110.653, 114.47, \
118.163, 121.745, 125.224, 128.609, 131.907, 135.125, 138.268, \
141.341, 144.348, 147.295, 150.183, 153.017, 155.799, 159.209, \
161.885}
```

```
Table[Sqrt[2*(psi)/(0.002297)], {psi, 1, 30, 1}]

{29.5076, 41.7301, 51.1087, 59.0153, 65.9811, 72.2787, 78.0699, \
83.4602, 88.5229, 93.3114, 97.8658, 102.217, 106.391, 110.407, \
114.283, 118.031, 121.663, 125.19, 128.621, 131.962, 135.221, \
138.403, 141.514, 144.557, 147.538, 150.46, 153.326, 156.14, 158.904, \
161.62}
```

Venturi

$$Q = A_2 * Cos\left(ArcSin\left(1 - \frac{\left(\frac{P_1 - P_2}{\rho\left(1 - \left(\frac{A_2}{A_1}\right)^2\right)} \right)}{c^2} \right) \right) * c$$

Worked example - Kerosene Flow through a Venturi Meter

The pressure difference $dp = p_1 - p_2$ between upstream and downstream is *100 kPa* $(1\ 10^5$ $N/m^2)$. The specific gravity of kerosene is *0.82*.

Upstream diameter is *0.1 m* and downstream diameter is *0.06 m*.

Density of kerosene can be calculated as:

$\rho = 0.82\ (1000\ kg/m^3)$

$= \underline{820}\ (kg/m^3)$

- Density, Specific Weight and Specific Gravity - An introduction and definition of density, specific weight and specific gravity. Formulas with examples.

Upstream and downstream area can be calculated as:

$A_1 = \pi \ ((0.1 \ m)/2)^2$

$\quad = \underline{0.00785} \ (m^2)$

$A_2 = \pi \ \ ((0.06 \ m)/2)^2$

$\quad = \underline{0.002826} \ (m^2)$

Theoretical flow can be calculated:

$q = A_2 \ [\ 2(p_1 - p_2) \ / \ \rho(1 - (A_2/A_1)^2) \ ]^{1/2}$

$q = (0.002826 \ m^2) \ [\ 2 \ (10^5 \ N/m^2) \ / \ (820 \ kg/m^3)(1 - (\ (0.002826 \ m^2) \ / \ (0.00785 \ m^2) \ )^2) \ ]^{1/2}$

$\quad = \underline{0.047} \ (m^3/s)$

For a pressure difference of 1 kPa (0,01x10^5 N/m^2) - the theoretical flow can be calculated:

$q = (0.002826 \ m^2) \ [\ 2 \ (0.01 \ 10^5 \ N/m^2) \ / \ (820 \ kg/m^3)(1 - (\ (0.002826 \ m^2) \ / \ (0.00785 \ m^2) \ )^2) \ ]^{1/2}$

$\quad = \underline{0.0047} \ (m^3/s)$

The mass flow can be calculated as:

$m = q \ \rho$

$\quad = (0.0047 \ m^3/s) \ (820 \ kg/m^3)$

$\quad = \underline{3.85} \ (kg/s)$

Mathematica

```
A1=0.00785
A2=0.002826
p=100000
d=820

q=A2*Sqrt[(2.0*p)/(d*(1-((A2/A1)*(A2/A1))))]

0.0473065
```

1.17 Simple Harmonic Motion solving for v using x and A

Harmonic Velocity, Amplitude A, position x, Spring force k, mass m.

$$v=\sqrt{\frac{k}{m}}\left(Cos\left(ArcSin\left(1-\left(\frac{1}{2}\frac{A^2}{c^2}-\frac{1}{2}\frac{x^2}{c^2}\right)\right)\right)*c\right)$$

TODO: Check if the ½ can be replaced by Cos(ArcSin(…)) combination.

Worked example

A=5 meters , x=2.5 meters, k=1 N/m ,m=2 N/m

Classic

Sqrt[(1.0/2.0)*((5.0*5.0) - (2.0*2.0))]
Result is 3.24037 m/s

Pi-Space

```
Sqrt[(1.0/2.0)]*
    (
        Cos[ArcSin[1 -
     (
      (
        ((5.0*5.0*0.5) - (2.0*2.0*0.5))/(299792458*299792458)
       )
      )
    ]]
  *299792458
 )
```

This produces a result of 3.15883 m/s. This is not the same as the classical result.

Pi-Space maintains that is a more "accurate" result.

To make the result match the Classical Result, we just need to make Speed of Light less accurate (incorrect!) e.g. 2997924

Let's redo the calculation

```
Sqrt[(1.0/2.0)]*
    (
        Cos[ArcSin[1 -
     (
      (
        ((5.0*5.0*0.5) - (2.0*2.0*0.5))/(2997924*2997924)
       )
      )
    ]]
  *2997924
```

)

This produces a result of 3.24038 m/s so they match.

This would need to be verified by experimentation. I do not have the equipment for this.

1.18 Average Transverse Kinetic Energy Due To Temperature, solving for velocity

Solving for Velocity in Pi-Space, we have the form

$$v = Cos\left(ArcSin\left(1 - \frac{\frac{3}{2}\frac{kT}{m}}{c^2} \right) \right) * c$$

Let’s solve a problem

Find Transverse KE and Average Velocity

T = 27 Degrees Celsius = 300 Kelvin
Mass Helium = 6.65*10^-27 Kg

Solve for Classic

KEtr = (3/2)∗(1.38∗10^−23)∗(300)= 6.21×10^{-21}

Solving for Velocity

Sqrt[(2.0*(6.2*10^-21))/(6.65*10^-27)]

V = 1365.53 = 1.37*10^3 m/J

Solve for Pi-Space

V =
(Cos[ArcSin[
    1 - (((6.21*10^-21)/(6.65*10^-27))/(299792458^2))]])*(299792458)
Gives us

V = 1366.63 = 1.37*10^3 m/J

1.19 Table of Formulas

Here we compare the Pi Space Theory Formulas versus the established formulas. These are Archimedean formulas in that they are calculated from the properties of Spheres.

Pi-Space units are v/c (atom diameter line change) and g/c^2 (atom area change) relative to Observer

Note: If you want to use these formulas with MPH or Meters, first convert the velocity value to v/c where c = 186000 mps for miles and c = 299,792,458 meters per second. Divide by 60*60 = 3600 if you want a per second value for your velocity. If you have an acceleration or a gravity value which are the same, divide by c^2 where c depends on the units you are working with. When you get a result from the formula and you want to convert back to your original units, if your units are 1/c (see formula "Units" column), then all you need to do is multiply the result by that value. If the units are 1/c^2 all you need to do is multiply by c^2. Energy has units 1/c^2 for example. Velocity has typically 1/c.

	Newton	**Einstein**	**Pi Space Theory (Brady)**	**Units**
Velocity Addition	u + v	u + v / 1 + uv		1/c
Velocity Subtraction	u - v	u - v / 1 – uv		1/c
Kinetic Energy	1/2mv^2		m*(1-Cos(ArcSin(v/c))) *c^2	1/c^2
Relativistic Kinetic Energy		mc^2/Sqrt(1 - v^2/c^2) - mc^2	mc^2 - mc^2*Sqrt(1 - v^2/c^2)	1/c^2
Total Energy		E=MC^2	E = M*Pi*C^2	1/c^2
Potential Energy	mgh		mgh	1/c^2
Velocity for KE=PE	mgh = 1/2mv^2		mgh = m*(1-Cos(ArcSin(v/c)))	1/c
Velocity for PE	v = Sqrt(2*gh)		v = Sin(ArcCos(1-gh/c^2))*c	1/c
Escape Velocity	v= Sqrt(2GMm /r)	Tuv = Guv	v = Sin(ArcCos(1-(GMm/r)/c^2))*c	1/c
Lorentz-Fitzgerald Transformation		Sqrt(1-v^2/c^2)	Cos(ArcSin(v/c))	1/c^2
Time Dilation		t = t' / Sqrt(1-v^2/c^2)	t = t' / Cos(ArcSin(v/c))	1/c
Distance Shortening		x = x' * Sqrt(1-v^2/c^2)	x = x' * Cos(ArcSin(v/c))	1/c^2
De Broglie Wavelength Shortening		(h/mv)*Sqrt(1-v^2/c^2)	(h/mv)*Cos(ArcSin(v/c))	1/c
Radius Excess		planet radius * GM/3c^2	planet radius * GM/c^2	1/c^2
Average Velocity	avg vel = v0 + v / 2		v = vo + v avg vel = (1-Cos(ArcSin(v/c))) / ArcSin(v/c)	1/c
Acceleration	v2 - v1 / t	Metric	(v2 - v1 / t) * gamma	

Gravity	Fg = GMm/r^2	Fg = G’Mm/Pi.r^2	Fg = GMm/r^2 (average acceleration or atom area change - multiply by "gamma" to calculate non-uniform value based on velocity range) Note: Full Pi-Space formula for Gravity is Fg = G’Mm/Pi*r^2 where Pi*r^2 represents the area of the "Planet's Gravity Pi Shell". Typically though Pi is ignored.	1/c^2
Non Uniform Acceleration Calculation "gamma"		Metric	Multiply Newtonian acceleration "a" by gamma value to get adjusted value. Input velocity range v1 to v2 into gamma formula Simple version (gamma measures changing slope of acceleration which is non constant) gamma = (v2 - v1) / (ArcSin(v2) - ArcSin(v1))	
Newtonian Acceleration	accel = v2 - v1 / t		accel = (v2 - v1 / t)*gamma	1/c^2
Distance traveled	s = vt + 1/2at^2		a1 = a * gamma (vo, vo + at) s = vt + (1-Cos(ArcSin(a1*t/c))/(ArcSin(a1*t/c))	1/c^2
Final velocity of falling object	v^2f = v^2o + 2ax		a1 = a * gamma (vo, vo + ax) vf = vo + Sin(ArcCos(1-(a1*x)/c^2))	1/c

Distance travelled at final velocity	h = vf^2/2g		g1 = g * gamma (vo, vf) h = (1 - (Cos(ArcSin(vf/c))) / (g1/c^2)	1/c^2
Solving for time t	See [1] below		Use Mathematica to solve for t, See [1] below	1/c
Black Hole Radius	r = 2GM/c^2		r = GM/c^2	1/c

	Bernoulli	**Pi-Space**	
Pitot - Velocity from Pressure	v=Sqrt(2*(Pt-Ps)/Rho)	v=Cos(ArcSin(1 – ((Pt-Ps/Rho)/c^2))*c	1/c
Venturi - Q - Flow	Q=A1*Sqrt(2*(Pt-Ps)/(Rho*(1-(A1/A2)^2))	Q=A1* Cos(ArcSin(1 – ((Pt-Ps/(Rho*(1-(A1/A2)^2)/c^2))*c	1/c

1.20 Navier-Stokes In Pi-Space

Flow e.g. xy area/energy (See Quantum Theory Doc)

$$\frac{\partial u}{\partial t} = \mu * \left(Cos\left(ArcSin\left(\frac{v}{c} \right) \right) \Psi(r,t) - \Psi(r,t) \right) + \frac{\left(\frac{p}{\rho} \right)}{c^2} \Psi(r,t) + \frac{\left(\frac{GM}{r^2} \right) h}{c^2} \Psi(r,t) + k \frac{(\nabla T)}{c^2} \Psi(r,t)$$

1.21 Navier Stokes Solving For Velocity (See Quantum Theory Doc)

For xy,yz and zx axis e.g.

$$FlowVelocity_{xy} = Cos\left(ArcSin\left(1 - \frac{\left(\frac{p}{\mu\rho} \right)}{c^2} - \frac{\frac{gh}{\mu}}{c^2} - \frac{\frac{k}{\mu}(\nabla T)}{c^2} - \frac{ExtTurb}{c^2} \right) \right) * c$$

Note: In the case of very small energy values, for example when we deal with an energy which is smaller than 10-17, the standard Java ArcSin() value fails. Therefore, we need to use the Square Root version of Cos(ArcSin(x)) which is Sqrt(1-x^2). This gives us

$$xtotal = \frac{\left(\frac{p}{\mu\rho}\right)}{c^2} - \frac{\frac{gh}{\mu}}{c^2} - \frac{\frac{k}{\mu}(\nabla T)}{c^2} - \frac{ExtTurb}{c^2}$$

Derivation

$$Cos(ArcSin(x)) = Sqrt(1 - x\text{^}2)$$

Replace x with xtotal

$$FlowVelocity_{xy} = Sqrt(1-(1-xtotal)\text{^}2.0)*c$$

$$FlowVelocity_{xy} = Sqrt(1-(1-2.0*xtotal + xtotal\text{^}2)*c$$

To solve for Velocity

$$FlowVelocity_{xy} = Sqrt(2.0*xtotal - xtotal\text{^}2)*c$$

1.22 Navier Stokes Solving For A Particular Value (See Quantum Theory Doc)

Let’s say we want to solve for Gravity or Pressure in Navier-Stoke. How can we do this in Pi-Space?

In Pi-Space, it’s reasonably straight forward.

We take any arbitrary values a,b,c representing various factors, totaling to s (which could be velocity). Let’s solve for a (which could be pressure).

Solve[Cos[ArcSin[1 - a - b - c]] == s, a]

{{a -> 1 - b - c - Sqrt[1 - s^2]}, {a -> 1 - b - c + Sqrt[1 - s^2]}}

{{a -> 1 - b - c - Cos[ArcSin[s]]}, {a -> 1 - b - c + Cos[ArcSin[s]]}}

So for an area loss to the Pi-Shell

a = 1 - b - c - Cos[ArcSin[s]]

1.23 Simple Harmonic Motion, Solving For Velocity knowing Amplitude, x, Spring constant k and mass m

(See Advanced Quantum Theory Doc)

$$v = \sqrt{\frac{k}{m}}\left(Cos\left(ArcSin\left(1 - \left(\frac{1}{2}\frac{A^2}{c^2} - \frac{1}{2}\frac{x^2}{c^2} \right) \right) \right) * c \right)$$

1.24 Average Transverse Kinetic Energy Due To Temperature, solving for velocity

$$v = Cos\left(ArcSin\left(1 - \frac{\frac{3}{2}\frac{kT}{m}}{c^2} \right) \right) * c$$

1.25 Ideal Gas Law

(See Temperature And Super Conductivity Doc)

$$\frac{\left(\frac{p}{\rho}\right)}{c^2}\Psi(r,t) = k\frac{(\nabla T)}{c^2}\Psi(r,t)$$

$$\frac{\left(\frac{p}{\rho}\right)}{c^2} = k\frac{(\nabla T)}{c^2}$$

This is the same as

$$pV = nkT$$

1.26 Charles Law Temperature And Volume In Pi-Space

(See Temperature And Super Conductivity Doc)

$$\left(\left(Cos\left(ArcSin\left(\frac{v}{c} \right) \right) \Psi(r,t) - \Psi(r,t) \right) + k\frac{(\nabla T)}{c^2}\Psi(r,t) \right)^{Local} \alpha volume$$

This is how we describe Charles' Law in Pi-Space.

1.27 Boyles Law Pressure And Volume In Pi-Space

(See Temperature And Super Conductivity Doc)

$$\left(\left(Cos\left(ArcSin\left(\frac{v}{c}\right)\right)\Psi(r,t)-\Psi(r,t)\right)+\frac{\left(\frac{pV}{m}\right)}{c^2}\Psi(r,t)\right)^{Local}$$

[1]

$$t = \frac{v_{0y}}{g} \pm \sqrt{\frac{v_{0y}^2}{g^2} - \frac{2h}{g}}$$

```
Solve[v*t + ((1-Cos[ArcSin[a]])/(ArcSin[a]))*(t*t) == s,t]
```

$$\left\{\left\{t \to \frac{-v\,ArcSin[a] - \sqrt{4\,s\,ArcSin[a] - 4\sqrt{1-a^2}\,s\,ArcSin[a] + v^2\,ArcSin[a]^2}}{2\left(1-\sqrt{1-a^2}\right)}\right\},\right.$$
$$\left.\left\{t \to \frac{-v\,ArcSin[a] + \sqrt{4\,s\,ArcSin[a] - 4\sqrt{1-a^2}\,s\,ArcSin[a] + v^2\,ArcSin[a]^2}}{2\left(1-\sqrt{1-a^2}\right)}\right\}\right\}$$

1.28 Lagrange (See Quantum Theory)

Joseph Louis Lagrange defined the energy of a system for the Path of Least Action as follows.

$$L = T - V$$

T is the Kinetic Energy

V is the potential Energy

$$L = T - V = \left(\rho * \left(Cos\left(ArcSin\left(\frac{v}{c}\right)\right)\Psi(r,t) - \Psi(r,t) + \frac{\left(\frac{GM}{r^2}\right)h}{c^2}\Psi(r,t)\right)\right)$$

1.29 Proper Time

$$\sqrt{1-\frac{v^2}{c^2}} = Cos(ArcSin(v/c))$$

This simplifies to

$$t * Cos(ArcSin(v/c))$$

$$50 * Cos(ArcSin(0.8))$$

30 light years proper time

1.30 Elastic Collisions in Pi-Space

Solving for individual values knowing v1 and v2, not knowing v1final and v2final.

```
Solve[(c*c*m1*(1 - Cos[ArcSin[v1]])) + (c*c*
  m2*(1 - Cos[ArcSin[v2]])) == (c*c*m1*(1 - Cos[ArcSin[v3]])) + (c*c*
  m2*(1 - Cos[ArcSin[((m1*v1 + m2*v2 - m1*v3)/m2)]])), v3]
```

```
v3 -> (m1^4 v1 - m2^4 v1 + 2 m1^3 m2 v2 - 2 m1 m2^3 v2 +
  2 m1^3 m2 v1^2 v2 + 2 m1 m2^3 v1^2 v2 + 6 m1^2 m2^2 v1 v2^2 +
  2 m2^4 v1 v2^2 + 4 m1 m2^3 v2^3 -
  2 m1^3 m2 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] +
  2 m1 m2^3 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] -
  2 m1^2 m2^2 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2] +
  2 m2^4 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2])/(m1^4 - 2 m1^2 m2^2 +
  m2^4 + 4 m1^2 m2^2 v1^2 + 4 m1^3 m2 v1 v2 + 4 m1 m2^3 v1 v2 +
  4 m1^2 m2^2 v2^2)
```

```
Solve[(c*c*m1*(1 - Cos[ArcSin[v1]])) + (c*c*
     m2*(1 - Cos[ArcSin[v2]])) == (c*c*
     m1*(1 - Cos[ArcSin[((m1 v1 + m2 v2 - m2 v4)/m1)]])) + (c*c*
     m2*(1 - Cos[ArcSin[v4]])), v4]
```

```
v4 -> (-2 m1^3 m2 v1 + 2 m1 m2^3 v1 + 4 m1^3 m2 v1^3 - m1^4 v2 +
    m2^4 v2 + 2 m1^4 v1^2 v2 + 6 m1^2 m2^2 v1^2 v2 +
    2 m1^3 m2 v1 v2^2 + 2 m1 m2^3 v1 v2^2 +
    2 m1^4 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] -
    2 m1^2 m2^2 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] +
    2 m1^3 m2 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2] -
    2 m1 m2^3 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2])/(m1^4 - 2 m1^2 m2^2 +
    m2^4 + 4 m1^2 m2^2 v1^2 + 4 m1^3 m2 v1 v2 + 4 m1 m2^3 v1 v2 +
    4 m1^2 m2^2 v2^2)
```

Worked example

Classic Formula Example

For example:

Ball 1: mass = 3 kg, velocity = 4 m/s

Ball 2: mass = 5 kg, velocity = −6 m/s

After collision:

Ball 1: velocity = −8.5 m/s

Ball 2: velocity = 1.5 m/s

Using the Pi-Space Approach with Mathematica

```
c = 186000

m1 = 3.0

m2 = 5.0

v1 = 4.0/c

v2 = -6.0/c


Solve[(m1*c*c* (1 - Cos[ArcSin[v1]])) + (m2*c*
   c *(1 - Cos[ArcSin[v2]])) == (m1*c*
```

```
c *(1 - Cos[ArcSin[v3]])) + (m2*c*
c*(1 - Cos[ArcSin[((m1*v1 + m2*v2 - m1*v3)/m2)]])), v3]
```

```
v3 = (m1^4 v1 - m2^4 v1 + 2 m1^3 m2 v2 - 2 m1 m2^3 v2 +
  2 m1^3 m2 v1^2 v2 + 2 m1 m2^3 v1^2 v2 + 6 m1^2 m2^2 v1 v2^2 +
  2 m2^4 v1 v2^2 + 4 m1 m2^3 v2^3 -
  2 m1^3 m2 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] +
  2 m1 m2^3 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] -
  2 m1^2 m2^2 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2] +
  2 m2^4 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2])/(m1^4 - 2 m1^2 m2^2 +
  m2^4 + 4 m1^2 m2^2 v1^2 + 4 m1^3 m2 v1 v2 + 4 m1 m2^3 v1 v2 +
  4 m1^2 m2^2 v2^2)
```

To get back to Newton velocity we need to multiply by speed of light. So multiply v3 by c.

```
v3 = (m1^4 v1 - m2^4 v1 + 2 m1^3 m2 v2 - 2 m1 m2^3 v2 +
  2 m1^3 m2 v1^2 v2 + 2 m1 m2^3 v1^2 v2 + 6 m1^2 m2^2 v1 v2^2 +
  2 m2^4 v1 v2^2 + 4 m1 m2^3 v2^3 -
  2 m1^3 m2 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] +
  2 m1 m2^3 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] -
  2 m1^2 m2^2 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2] +
```

2 m2^4 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2]) / (m1^4 - 2 m1^2 m2^2 +

m2^4 + 4 m1^2 m2^2 v1^2 + 4 m1^3 m2 v1 v2 + 4 m1 m2^3 v1 v2 +

4 m1^2 m2^2 v2^2)*c

This produces -8.5 which matches the Classic Formula

```
In[7]:= c = 186000
        m1 = 3.0
        m2 = 5.0
        v1 = 4.0/c
        v2 = -6.0/c

        v3 =
         (m1^4 v1 - m2^4 v1 + 2 m1^3 m2 v2 - 2 m1 m2^3 v2 + 2 m1^3 m2 v1^2 v2 + 2 m1 m2^3 v1^2 v2 + 6 m1^2 m2^2 v1 v2^2 + 2 m2^4 v1 v2^2 +
              4 m1 m2^3 v2^3 - 2 m1^3 m2 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] + 2 m1 m2^3 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] -
              2 m1^2 m2^2 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2] + 2 m2^4 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2]) /
           (m1^4 - 2 m1^2 m2^2 + m2^4 + 4 m1^2 m2^2 v1^2 + 4 m1^3 m2 v1 v2 + 4 m1 m2^3 v1 v2 + 4 m1^2 m2^2 v2^2) * c

Out[7]= 186000

Out[8]= 3.

Out[9]= 5.

Out[10]= 0.0000215054

Out[11]= -0.0000322581

Out[12]= -8.5
```

v4 =(-2 m1^3 m2 v1 + 2 m1 m2^3 v1 + 4 m1^3 m2 v1^3 - m1^4 v2 +

m2^4 v2 + 2 m1^4 v1^2 v2 + 6 m1^2 m2^2 v1^2 v2 +

2 m1^3 m2 v1 v2^2 + 2 m1 m2^3 v1 v2^2 +

2 m1^4 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] -

2 m1^2 m2^2 v1 Sqrt[1 - v1^2] Sqrt[1 - v2^2] +

2 m1^3 m2 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2] -

2 m1 m2^3 Sqrt[1 - v1^2] v2 Sqrt[1 - v2^2])/(m1^4 - 2 m1^2 m2^2 +

m2^4 + 4 m1^2 m2^2 v1^2 + 4 m1^3 m2 v1 v2 + 4 m1 m2^3 v1 v2 +

4 m1^2 m2^2 v2^2)*c

This produces 1.5 which is also correct.

Note: A result of 0 means that we have an elastic collision where both mass and velocity of collision are equal so v1'=v2 and v2'=v1.

1.31 Inelastic Collisions In Pi-Space

Let's solve this in Classic and Pi-Space.

Classic

```
In[1]:= Solve[(0.5 * m1 * v1 * v1) + (0.5 * m2 * v2 * v2) ==
          excessEnergy +
           ((((m1 * v1 + m2 * v2) / (m1 + m2))) * (((m1 * v1 + m2 * v2) / (m1 + m2))) *
             (m1 + m2) * 0.5), excessEnergy]
```

$$\text{Out[1]=}\ \left\{\left\{\text{excessEnergy} \to -1.\left(-0.5\,\text{m1}\,\text{v1}^2 - 0.5\,\text{m2}\,\text{v2}^2 + \frac{0.5\,(\text{m1}\,\text{v1} + \text{m2}\,\text{v2})^2}{\text{m1} + \text{m2}}\right)\right\}\right\}$$

{{excessEnergy -> -1. (-0.5 m1 v1^2 - 0.5 m2 v2^2 + (
    0.5 (m1 v1 + m2 v2)^2)/(m1 + m2))}}

Pi-Space

CC is C^2
Also we use Abs[] for Velocity because we add up the energies.

```
In[2]:= Solve[
          (cc * m1 * (1 - Cos[ArcSin[Abs[v1]]])) +
            (cc * m2 * (1 - Cos[ArcSin[Abs[v2]]])) ==
           (lostEnergy) +
            (cc * (m1 + m2) * (1 - Cos[ArcSin[Abs[((m1 * v1 + m2 * v2) / (m1 + m2))]]])),
          lostEnergy]
```

$$\text{Out[2]=}\ \left\{\left\{\text{lostEnergy} \to -\text{cc m1}\sqrt{1-\text{Abs}[\text{v1}]^2} - \text{cc m2}\sqrt{1-\text{Abs}[\text{v2}]^2} + \text{cc m1}\sqrt{1-\text{Abs}\left[\frac{\text{m1 v1}+\text{m2 v2}}{\text{m1}+\text{m2}}\right]^2} + \text{cc m2}\sqrt{1-\text{Abs}\left[\frac{\text{m1 v1}+\text{m2 v2}}{\text{m1}+\text{m2}}\right]^2}\right\}\right\}$$

```
{{lostEnergy -> -cc m1 Sqrt[1 - Abs[v1]^2] -
   cc m2 Sqrt[1 - Abs[v2]^2] +
   cc m1 Sqrt[1 - Abs[(m1 v1 + m2 v2)/(m1 + m2)]^2] +
   cc m2 Sqrt[1 - Abs[(m1 v1 + m2 v2)/(m1 + m2)]^2]}}
```

Worked Example

Two objects 2300kg at 22 mph collides with 780kg at 26 mph. What is the energy loss?

Classic

```
c=186000
m1=2300
v1=22
m2=780
v2=-26.0
```

```
Solve[(0.5*m1*v1*v1)+(0.5*m2*v2*v2)==excessEnergy+((((m1*v1+m2*v2)/(m1+m2)))*(((m1*v1+m2*v2)/(m1+m2)))*(m1+m2)*0.5),excessEnergy]
```

```
{{excessEnergy→671003.}}
```

Pi-Space

```
c=186000
m1=2300
v1=22/c
m2=780
v2=-26.0/c
cc=c*c
```

Solve[(cc*m1*(1−Cos[ArcSin[Abs[v1]]]))+(cc*m2*(1−Cos[ArcSin[Abs[v2]]]))==(lostEnergy)+(cc*(m1+m2)*(1−Cos[ArcSin[Abs[((m1*v1+m2*v2)(m1+m2))]]])),lostEnergy]

```
{{lostEnergy→671003.}}
```

Core Pi Space Math Ideas

Formula	**Existing**	**Pi Space**
Sphere addition	c^2 = a^2 + b^2	Pi*c^2 = Pi*a^2 + Pi*b^2
Atom Area		Pi*d^2
Diameter size of Observer atom	C	1
Atom diameter loss due to movement	Velocity	Sin(ArcSin(v/c)) = v/c
Remaining diameter due to movement	Lorentz-Fitz	Cos(ArcSin(v/c))
General movement equation, Law of the Cosines	c^2 = a^2 + b^2 + 2abCos(Theta)	Pi.c^2 = Pi.a^2 + Pi.b^2 + 2.Pi.a.b.Cos(Theta)
General angle equation for interacting atoms, Law of the Sines	a/Sin(a) = b/Sin(b) = c/Sin(c)	a/Sin(a) = b/Sin(b) = c/Sin(c)
Range of velocities	0..C	0..ArcSin(v/c))
Temperature	Kelvin	Wave Amplitude maps to Kelvin
Electricity and Gravity	n/a	e^ix EG = cos x + i sin x
Magnetic and Turbulence	n/a	e^ix MT = cos x + i sin x perpendicular to e^ix EG